User manual for

Copy Number Variant Detection Plugin (beta)

Windows, Mac OS X and Linux

January 8, 2015

This software is for research purposes only.

CLC bio, a QIAGEN Company
Silkeborgvej 2
Prismet
DK-8000 Aarhus C
Denmark

Contents

1 Introduction to the *Copy Number Variant Detection Plugin* **4**

2 Running the *Copy Number Variant Detection Plugin* **6**

2.1 Algorithm and parameter description 6

How to set the fold-change cutoff when the sample purity is not 100% . . . 8

2.2 Outputs of the Copy Number Variant Detection tool 10

2.2.1 Region-level CNV track 10

How to interpret fold-changes when the sample purity is not 100% 11

2.2.2 Target-level CNV track 12

2.2.3 Gene-level annotation track 14

2.2.4 Report 15

Normalization 15

Target-level log2-ratios 15

CNV statistics 16

2.2.5 Algorithm report 16

Normalization and chromosome analysis 16

Prediction of target-level CNVs 17

Prediction of region-level CNVs 18

3 Installation **20**

4 Uninstall **22**

Bibliography 23

Chapter 1

Introduction to the *Copy Number Variant Detection Plugin*

The *Copy Number Variant Detection Plugin* is designed to detect copy number variations (CVNs) from targeted resequencing experiments. The current release is a beta version.

The tool takes read mappings and target regions as input, and produces amplification and deletion annotations. The annotations are generated by a "depth-of-coverage" method, where the target-level coverages of the case and the controls are compared in a statistical framework. The algorithm implemented in *Copy Number Variant Detection Plugin* is inspired by the following papers:

- **Li et al.**, *CONTRA: copy number analysis for targeted resequencing*, Bioinformatics. 2012, 28(10):1307-1313 [Li et al., 2012].
- **Niu and Zhang**, *The screening and ranking algorithm to detect DNA copy number variations*, Ann Appl Stat. 2012, 6(3): 1306-1326 [Niu and Zhang, 2012].

The Copy Number Variant Detection tool identifies CNVs regions where the normalized coverage is statistically significantly different from the controls.

The algorithm carries out the analysis in several steps.

1. Base-level coverages are analyzed for all samples, and a robust coverage baseline is generated using the control samples.
2. Chromosome-level coverage analysis is carried out on the case sample, and any chromosomes with unexpectedly high or low coverages are identified.
3. Sample coverages are normalized, and a global, target-level statistical model is set up for the variation in fold-change as a function of coverage in the baseline.
4. Each chromosome is segmented into regions of similar fold-changes.
5. The expected fold-change variation in region is determined using the statistical model for target-level coverages. Region-level CNVs are identified as the regions with fold-changes significantly different from 1.0.

6. If chosen in the parameter steps, gene-level CNV calls are also produced.

Chapter 2

Running the *Copy Number Variant Detection Plugin*

2.1 Algorithm and parameter description

To start the Copy Number Variant Detection tool, click:

Toolbox | **Resequencing Analysis (** **)** | **Copy Number Variant Detection**

Select the case read mapping and click **Next**. You are now presented with choices regarding the data to use in the CNV prediction method, as shown in figure 2.1.

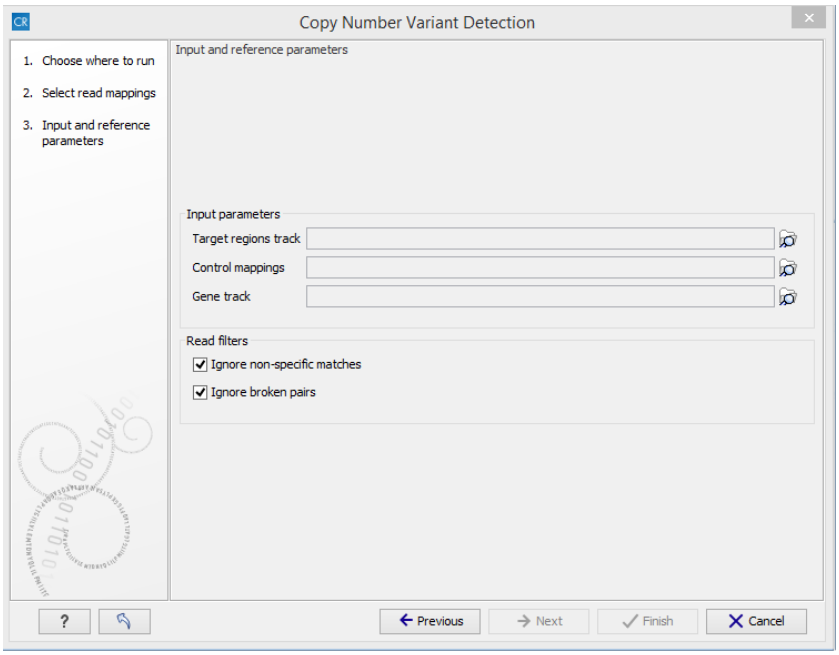

Figure 2.1: *The first step of the CNV detection tool.*

Target regions track An annotation track containing the regions targeted in the experiment must be chosen. This track must not contain overlapping regions, or regions made up of several intervals, because the algorithm is designed to operate on simple genomic regions.

Control mappings You must specify one or more read mappings, which will be used to create a baseline by the algorithm. For the best results, the controls should be matched with respect to the most important experimental parameters, such as gender and technology. If using non-matched controls, the CNVs reported by the algorithm may be less accurate.

Gene track Optional: If you wish, you can provide a gene track, which will be used to produce gene-level output as well as CNV-level output.

Ignore non-specific matches If checked, the algorithm will ignore any non-specifically mapped reads when counting the coverage in the targeted positions. Note: If you are interested in predicting CNVs in repetitive regions, this box should be unchecked.

Ignore broken pairs If checked, the algorithm will ignore any broken paired reads when counting the coverage in the targeted positions.

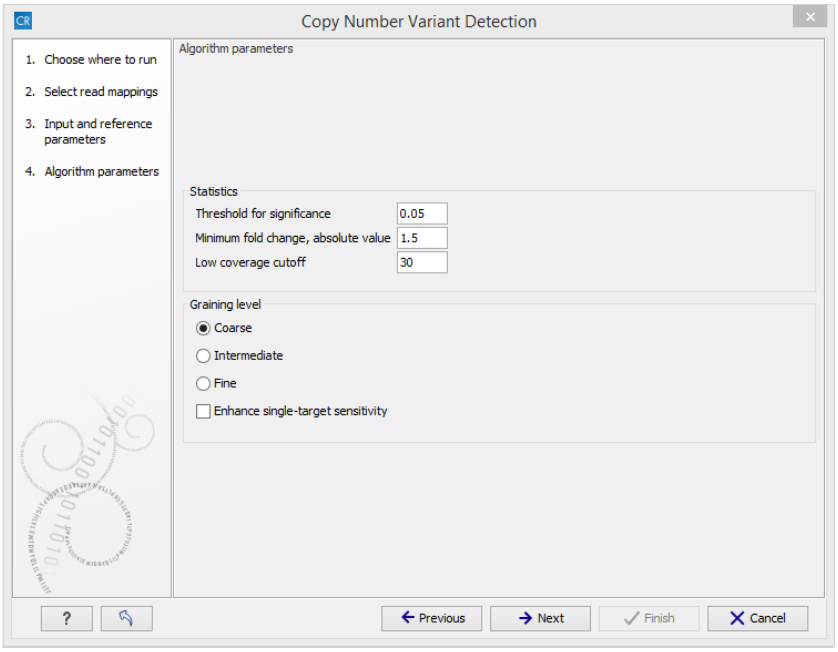

Figure 2.2: *The second step of the CNV detection tool*

Click **Next** to set the parameters related to the target-level and region-level CNV detection, as shown in as shown in figure 2.2.

Threshold for significance P-values lower than the threshold for significance will be considered "significant". The higher you set this value, the more CNVs will be predicted.

Minimum fold change, absolute value You must specify the minimum fold change for a CNV call. If the absolute value of the fold change of a CNV is less than the value specified in this parameter, then the CNV will be filtered from the results, even if it is otherwise statistically significant. For example, if a minimum fold-change of 1.5 is chosen, then the adjusted coverage of the CNV in the case sample must be either 1.5 times higher or 1.5 times lower than the coverage in the baseline, for it to pass the filtering step. If you do not want to filter on the fold-change, enter 0.0 in this field. Note: If your sample purity is less than 100%, it is necessary to take that into account when you adjust the fold-change cutoff. This is described in more detail in section 2.1.

Low coverage cutoff If the average coverage of a target is below this value, it will be considered "low coverage" and it will not be used to set up the statistical models, and p-values will not be calculated for it in the target-level CNV prediction.

Graining level The graining level is used for the region-level CNV prediction. Coarser graining levels produce longer CNV calls and less noise, and the algorithm will run faster. However, smaller CNVs consisting of only a few targets may be missed at a coarser graining level.

- Coarse: prefers CNVs consisting of many targets. The algorithm is most sensitive to CNVs spanning over 10 targets. This is the recommended setting if you expect large-scale deletions or insertions, and want a minimal false positive rate.
- Intermediate: prefers CNVs consisting of an intermediate number of targets. The algorithm is most sensitive to CNVs spanning 5 or more targets. This is the recommended setting if you expect CNVs of intermediate size.
- Fine: prefers CNVs consisting of fewer targets. The algorithm is most sensitive to CNVs spanning 3 or more targets. This is the recommended setting if you want to detect CNVs that span just a few targets, but the false positive rate may be increased.

Note: The CNV sizes listed above are meant as general guidelines, and are not to be interpreted as hard rules. Finer graining levels will produce larger CNVs when the signals for this are sufficiently clear in the data. Similarly, the coarser graining levels will also be able to predict shorter CNVs under some circumstances, although with a lower sensitivity.

Enhance single-target sensitivity All of the graining levels assume that a CNV spans more than one target. If you are also interested in very small CNVs that affect down to a single target in your data, check the 'Enhance single-target sensitivity' box. This will increase the sensitivity of detection of very small CNVs, and has the greatest effect in the case of the coarser graining levels. Note however that these small CNV calls are much more likely to be false positives. If this box is unchecked, only larger CNVs supported by several targets will be reported, and the false positive rate will be lower.

Clicking **Next**, you are presented with options about the results (see figure 2.3). In this step, you can choose to create an algorithm report by checking the **Create algorithm report** box. Furthermore, you can choose to output results for every target in your input, by checking the **Create target-level CNV track** box.

When finished with the settings, click **Next** to start the algorithm.

How to set the fold-change cutoff when the sample purity is not 100%

Given a sample purity of $X\%$, and a desired detection level (absolute value of fold-change in 100% pure sample) of T, the following formula gives the required fold-change cutoff:

$$\text{cutoff} = \frac{X\%}{100\%} \times T + (1 - \frac{X\%}{100\%}) \tag{2.1}$$

For example, if the sample purity is 40%, and you want to detect 6-fold amplifications or deletions (e.g. 12 copies instead of 2, or 2 copies instead of 12), then the cutoff should be:

$$\text{cutoff} = \frac{40\%}{100\%} \times 6 + (1 - \frac{40\%}{100\%}) = 3.0. \tag{2.2}$$

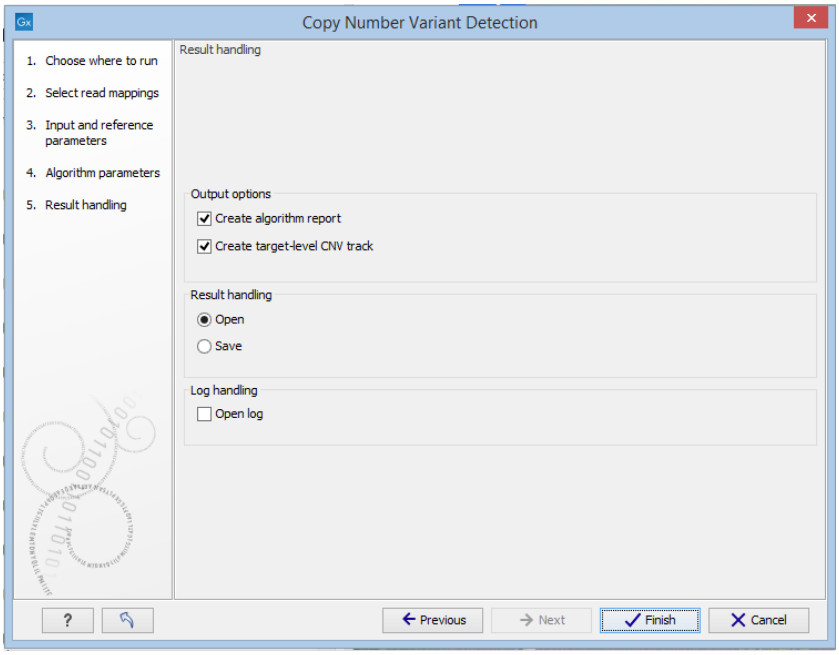

Figure 2.3: *Specifying whether an algorithm report and a target-level CNV track should be created.*

Figures 2.4 shows the required fold-change cutoffs in order to detect a particular degree of amplification/deletion at different sample purities. Figure 2.5 zooms in for low-level amplifications and deletions.

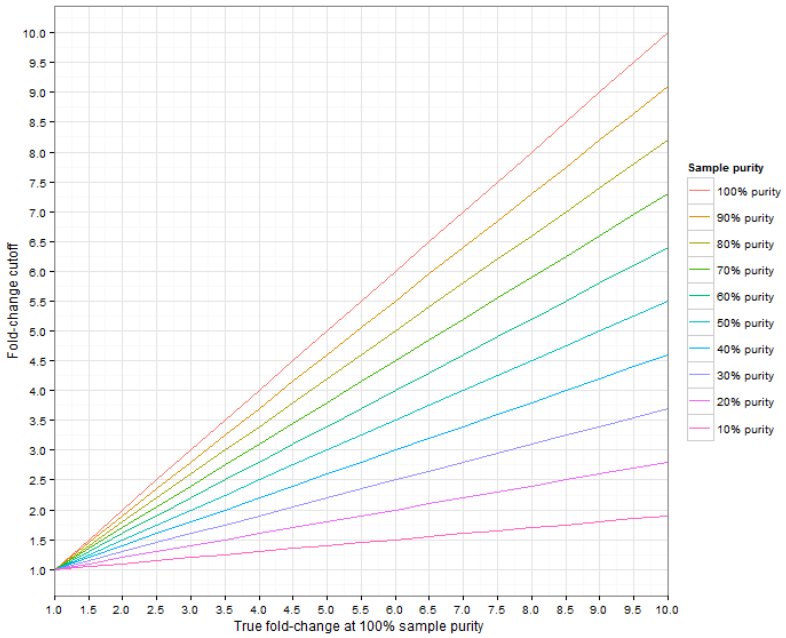

Figure 2.4: *The required fold-change cutoff to detect amplifications and deletions of different magnitudes, as a function of sample purity.*

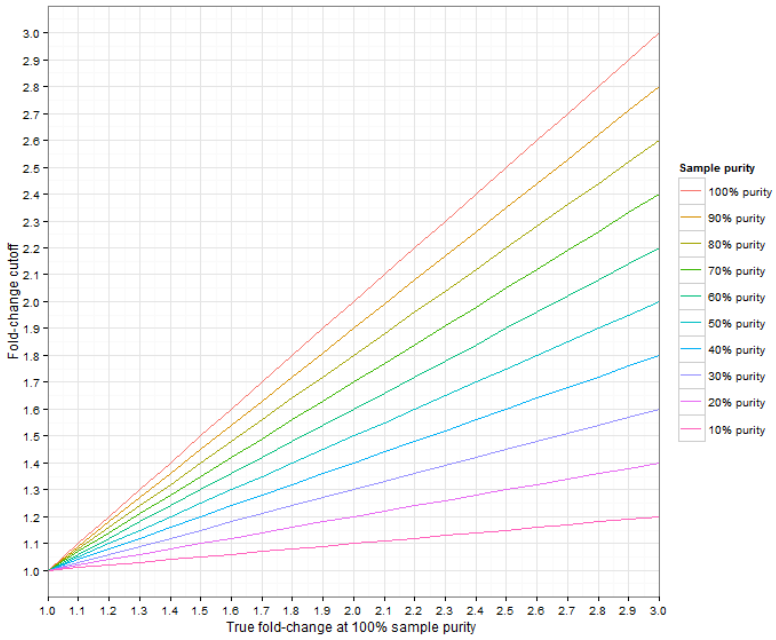

Figure 2.5: *Low-level amplifications and deletions: The required fold-change cutoff to detect amplifications and deletions of different magnitudes, as a function of sample purity.*

2.2 Outputs of the Copy Number Variant Detection tool

The Copy Number Variant Detection tool calls CNVs that are both global outliers on the target-level, and locally consistent on the region-level. The tool produces several outputs, which are described below.

2.2.1 Region-level CNV track

The algorithm will produce a region-level annotation track, which contains the CNV regions detected by the algorithm. Every annotation in this track joins one more more targets from the input target track, to produce contiguous CNVs. Each CNV in the region-level tracks is characterized in terms of the following properties:

Minimum CNV length: The minimum CNV length is the length of the region-level CNV annotation. This number should be interpreted as the lowest bound for the size of the CNV. The "true" CNV can extend into the adjacent genomic regions that have not be targeted.

P-value: The p-value corresponds to the probability that an observation identical to the CNV, or even more of an outlier, would occur by chance under the null hypothesis. The null hypothesis is that of no CNVs in the data. The p-value for a CNV region is calculated by combining the p-values of its constituent targets (discarding any low-coverage targets).

Fold-change (adjusted): The fold-change of the *adjusted* case coverage compared to the baseline. Negative fold-changes indicate deletions, and positive fold-changes indicate amplifications. A fold-change of 1.0 (or -1.0) represents identical coverages. The fold-changes are adjusted for statistical differences between targets with different sequencing depths. Note: if your sample purity is less than 100%, you need to take that into account when interpreting the fold-change values. This is described in more detail in section 2.2.1.

Consequence: The consequence classifies statistically significant CNVs as "Gain" or "Loss".

Number of targets: The total number of targets forming the (minimal) CNV region.

Targets: A list of the names of the targets forming the (minimal) CNV region. Note however that the list is truncated to 100 characters. If you want to see all the targets that constitute the CNV region, you can use the target-level output (section 2.2.2).

Comments: The comments can include useful information for interpreting individual CNV calls. The possible comments are:

1. Small region: If a region only consists of 1 target, it is classified as a 'small region'. The p-value of this region is therefore based on evidence from just one target, and may be less accurate than p-values for larger regions.
2. Disproportionate chromosome coverage: If a region is found on a chromosome that was determined to have disproportionate coverage, this will be noted in the comments. This means that the targets constituting this region were not used to set up the statistical models. Furthermore, the size and fold-change value of this CNV region may explain why the chromosome was detected to have disproportionate coverage.
3. Low coverage: If all targets inside a region had low-coverage, then the region will be classified as a 'low-coverage' region, and will be given a p-value of 1.0. You will only see these regions in the results if you set the significance cutoff to 1.0.

These properties can be found in separate columns when viewing the tracks in table view.

Note: The region-level calls do not guarantee that a single, larger CNV will always be called in just one CNV region. This is because adjacent region-level CNV calls are not joined into a single region if their average fold-changes are sufficiently different. For example, if a 2-fold gain is detected in a region and a 3-fold gain is detected in an immediately adjacent region of equal size, then these may appear in the results as two separate CNVs, or one single CNV with a 2.5-fold gain, depending on your chosen graining level, and the fold-changes observed in the rest of the data.

How to interpret fold-changes when the sample purity is not 100%

If your sample purity is less than 100%, it is necessary to take that into account when interpreting the fold-change values. Given a sample purity of X%, and an observed fold-change of F, following formula gives the actual fold-change that would be seen if the sample were 100% pure:

$$\text{fold-change in 100\% pure sample} = \frac{F-1}{X/100\%} + 1 \tag{2.3}$$

For example, if the sample purity is 40%, and you have observed a fold-change of 3, then the fold-change in the 100% pure sample would have been:

$$\text{fold-change in 100\% pure sample} = \frac{3.0-1}{40\%/100\%} + 1 = 6.0. \tag{2.4}$$

Figures 2.6 shows the 'true' fold changes for different observed fold-changes at different sample purities. Figure 2.7 zooms in for low-level amplifications and deletions.

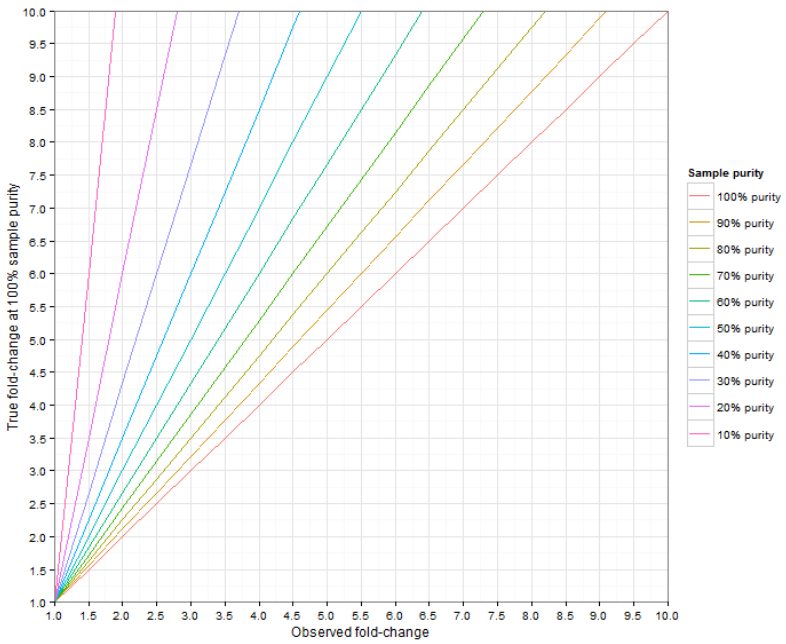

Figure 2.6: *The true fold-change in the 100% pure sample, for different observed fold-changes, as a function of sample purity.*

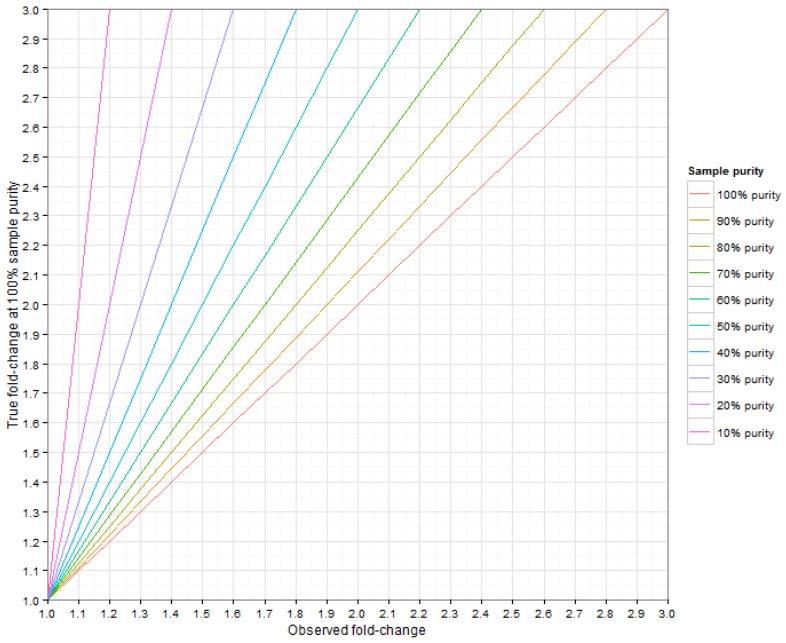

Figure 2.7: *Low-level amplifications and deletions: the true fold-change in the 100% pure sample, for different observed fold-changes, as a function of sample purity.*

2.2.2 Target-level CNV track

The algorithm will produce a target-level CNV track, if you've chosen to create one when running the algorithm. The target-level CNV track is an annotation track, containing one annotation for every target in the input data. Inspection of the target-level CNV track can give you additional information about both the CNVs called in the region-level results, and those regions that have not been called. Each target is annotated with the following information:

Target number: Targets are numbered in the order in which they occur in the genome. This information is used by the results report (see section 2.2.4).

Case coverage: The normalized coverage of the target in the case sample.

Baseline coverage: The normalized coverage of the target in the baseline.

Length: The length of the target region.

P-value: The p-value corresponds to the probability that an observation identical to the CNV, or even more of an outlier, would occur by chance under the null hypothesis. The null hypothesis is that of no CNVs in the data. The p-value in the target-level output reflects the global evidence for a CNV at that particular target. The target-level p-values are combined to produce the region-level p-values in the region-level CNV output.

FDR-corrected p-value: The FDR-corrected p-values correct for false positives arising from carrying out a very high number of statistical tests. The FDR-corrected p-value will, therefore, always be larger than the uncorrected p-value.

Fold-change (raw): The fold-change of the normalized case coverage compared to the normalized baseline coverage. The normalization corrects for the effects of different library sizes between the different samples. Negative fold-changes indicate deletions, and positive fold-changes indicate amplifications. A fold-change of 1.0 represents identical coverages.

Fold-change (adjusted): As observed by Li et al (2012, [Li et al., 2012]), the fold-changes (raw) depend on the coverage. Therefore, the fold-changes have to be adjusted for statistical differences between targets with different sequencing depths, before the statistical tests are carried out. The results of this adjustment are found in the ‘‘Fold-change (adjusted)’’ column. Note that sometimes, this will mean that a change that appears to be an amplification in the ‘‘raw’’ fold-change column may appear to be a deletion in the ‘‘adjusted’’ fold-change column, or vice versa. This is simply because for a given coverage level, the raw fold-changes were skewed towards amplifications (or deletions), and this effect was corrected in the adjustment. Note: if your sample purity is less than 100%, you need to take that into account when interpreting the fold-change values. This is described in more detail in section 2.2.1.

Region (joined targets): The region to which this target was classified to belong. The region may or may not have been predicted to be a CNV.

Regional fold-change: The adjusted fold-change of the region to which this target belongs. This fold-change value is computed from all targets constituting the region.

Regional p-value: The p-value of the region to which this target belongs. This is the p-value calculated from combining the p-values of the individual targets inside the region.

Regional consequence: If the target is included in a CNV region, this column will show "Gain" or "Loss", depending on the direction of change detected for the region. Note, however, that the change detected for the region may be inconsistent with the fold-change for a single target in the region. The reason for this is typically statistical noise at the single target.

Regional effect size: The effect size of a target-level CNV reflects the magnitude of the observed fold-change of the CNV region in which the target was found. The effect size of a CNV is classified into the following categories: "Strong" or "Weak". The effect size is "Strong" if

the fold-change exceeds the fold-change cutoff specified in the parameter steps. Otherwise, the effect size will be "Weak". Note, however, that "Weak" CNV calls will be filtered from the region-level output.

Comments: The comments can include useful information for interpreting the CNV calls. Possible comments in the target-level output are:

1. Low coverage target: If the target had a coverage under the specified coverage cutoff, it will be classified as low-coverage. Low-coverage targets were not used in calculating the statistical models, and will not have p-values.
2. Disproportionate chromosome coverage: If the target occurred on a chromosome that was detected to have disproportionate coverage. In this case, the target was not used to set up the statistical models.
3. Atypical fold-change in region: If there is a discrepancy between the direction of fold-change detected for the target and the direction of fold-change detected for the region, then the fold-change of the target is "atypical" compared to the region. This is usually due to statistical noise, and the regional fold-change is likely to be more accurate in the interpretation, especially for large regions.

2.2.3 Gene-level annotation track

If you have specified a gene track in the input parameters, you will get a gene-level CNV track as well. The gene-level CNV track is an annotation track, which is obtained by intersecting the region-level CNV track with the gene track in the input (ignoring any genes that do not overlap with the targets). Note that a single CNV may be reported several times in different genes, and a single gene may also be reported several times, if it is affected by more than one CNV. In addition to the annotations on the gene track supplied in the input parameters, the gene-level CNV track contains the following annotation columns:

Region length: The length of the actual annotation. That is, the length of the CNV region intersected with the gene.

CNV region: The *entire* CNV region affecting this gene (and possibly other genes).

CNV region length: The length of the *entire* CNV region affecting this gene (and possibly other genes).

Consequence: The consequence classifies statistically significant CNVs as "Gain" or "Loss".

Fold-change (adjusted): The adjusted fold-change of the *entire* CNV region affecting this gene (and possibly other genes).

P-value: The p-value of the *entire* CNV region affecting this gene (and possibly other genes).

Number of targets: The total number of targets forming the *entire* CNV region affecting this gene (and possibly other genes).

Comments: If the CNV region affecting this gene had any comments (as described in section 2.2.1), this will be present in the gene-level results as well.

Targets: A list of the names of the targets forming the (minimal) CNV region forming the *entire* CNV region affecting this gene (and possibly other genes). Note however that the list is truncated to 100 characters. If you want to know the full list of targets inside the CNV region, you can use the target-level output track.

2.2.4 Report

The report contains information about the results of the Copy Number Variant Detection tool. The sections of this report are described individually below.

Normalization

The Normalization section gives information about the sample-level and chromosome-level coverages. Any chromosomes with disproportionate coverages are noted; targets on these chromosomes were ignored when setting up the statistical models.

Target-level log2-ratios

The target-level coverage log-ratios are presented as a graph. An example is shown in figure 2.8. On the horizontal axis, the targets are placed in the order in which they appear in the genome. On the vertical axis, the adjusted coverage log-ratio of each target is plotted. The black line represents the actually observed mean adjusted log-ratio of coverage for each target. The cyan and red lines represent the 95% confidence intervals of the expected mean adjusted log-ratios of coverages, based on the statistical model. Chromosome boundaries are indicated as vertical lines.

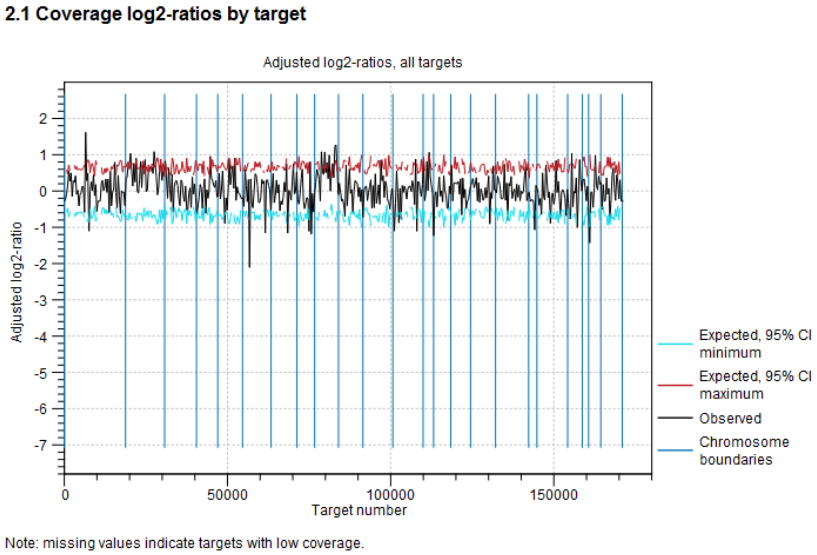

Figure 2.8: *An example graph showing the mean adjusted log-ratios of coverages in the report produced by the Copy Number Variant Detection tool. In this example, the second and ninth chromosomes are amplified, and the log-ratios of coverages of targets on these chromosome are significantly higher than for targets on other chromosomes. The black line in these regions is outside the boundaries defined by the cyan and red lines.*

CNV statistics

The last section in the report provides some information about the number of CNVs called in the region-level prediction results. The number of uncalled or filtered regions are also shown.

2.2.5 Algorithm report

If you have chosen to produce an algorithm report in the output handling step of the wizard, an algorithm report will also be produced. This contains information about the statistical models of the algorithm, and can be used to evaluate how well the assumptions of the model were fulfilled. We will now present the different sections of this report.

Normalization and chromosome analysis

This section of the report is related to the first step of the Copy Number Variant Detection tool, where the chromosome-level coverages are analyzed to detect any outliers. The total coverages of the case chromosomes are plotted against the total coverages of the baseline, and the detected outliers are indicated. Chromosome coverages identified as disproportionate are marked with red crosses (see figure 2.9).

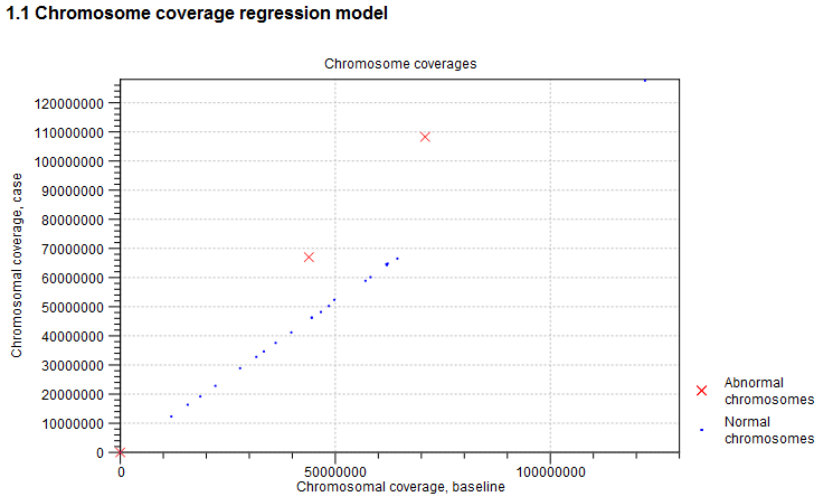

Figure 2.9: *An example graph showing the coverages of the chromosomes in the case versus the baseline. In this example, three chromosomes are marked as abnormal. Two of these chromosomes are significantly amplified, and log-ratios of coverages of many targets on these chromosome are significantly higher than for targets on other chromosomes. The third outlier chromosome had zero coverage in both the case and the baseline.*

The graph is followed by a table, where the detailed chromosome coverages are shown after normalization. Chromosomes with disproportionate coverage and chromosomes without any targets are marked in the 'Comment' column. These chromosomes are the ones marked with red crosses in the graph in section 1.1. of the algorithm report, and these chromosomes were not used in the coverage normalization step.

Prediction of target-level CNVs

This section of the algorithm report gives information about the statistical models used to predict target-level CNVs.

Adjustment of log2-ratios The first two graphs in this section are related to the adjustment of the log-ratios of coverages as a function of log-coverage. The log-ratio of coverages for targets depends on the level of coverage of the target, as observed by Li et al. (Bioinformatics, 2012), who also proposed that a linear correction should be applied [Li et al., 2012]. In the first of the two graphs, the non-adjusted log-ratios of target coverages are plotted against the log-coverage of the targets. In the second graph, the mean log-ratios are plotted after adjustment (figure 2.10). If the model fits the data, we expect to see that the adjusted mean log-ratios are centered around 0 for all log-coverages, and the variation decreases with increasing log-coverage.

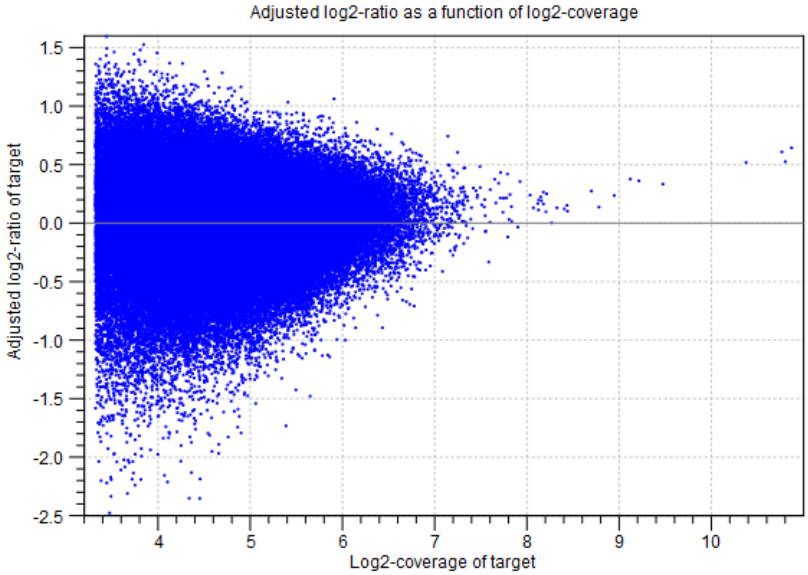

Figure 2.10: *An example graph showing the mean adjusted log-ratios of coverages plotted against the log-coverages of targets, in the algorithm report of the Copy Number Variation Detection plugin. Here, the adjusted mean log-ratios are centered around 0.0 for most coverages, and the variation decreases with increasing log-coverage. This indicates a good fit of the model. However, at very high coverages, the adjusted log-ratios are centered higher than 0.0, which indicates that for these coverages, the model is not a perfect fit. But only very few targets are affected by this, as the points are very sparse at these high coverage levels.*

Statistical model for adjusted log2-ratios In this section of the algorithm report, you can see how well the algorithm was able to model the statistical variation in the log-ratios of coverages. An example is shown in figure 2.11). A good fit of the model to the data points indicates that the variance has been modelled accurately.

To make the points more visible, double-click the figure, to open it in a separate editor. Here, you can select how to visualize the data points and the fitted model. For example, you can choose to highlight the data points in the sidepanel:

MA Plot Settings | **Dot properties** | **Dot type** | "Dot"

Distribution of adjusted log2-ratios in bins One of the assumptions of the statistical model used by the CNV detection tool is that the coverage log-ratios of targets are normally distributed with a

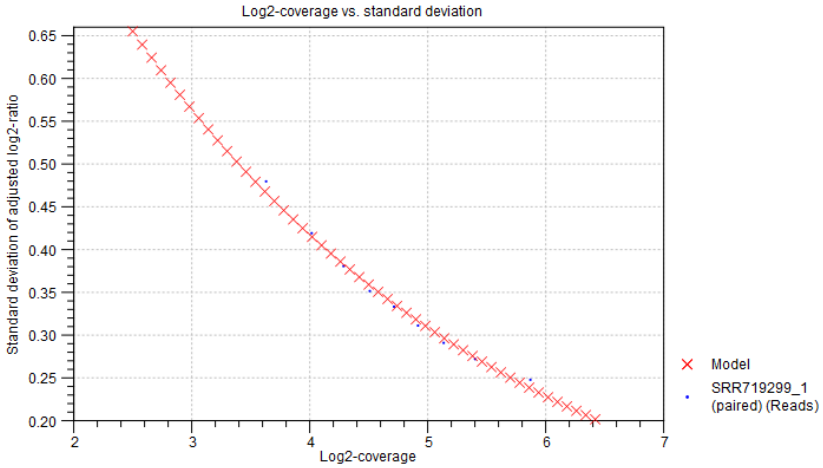

Figure 2.11: *An example graph showing how the variance in the target-level mean log-ratios was modelled in the algorithm report of the Copy Number Variation Detection plugin. Here, the data points are very close to the fitted model, indicating a good fit of the model to the data.*

mean of zero, and the variance only depends on the log-coverage of each target in the baseline. The bar charts in this section of the algorithm report show how well this assumption of the model fits the data. An example is shown in figure 2.12). A good fit of the model to the data points indicates that the variance has been modelled accurately.

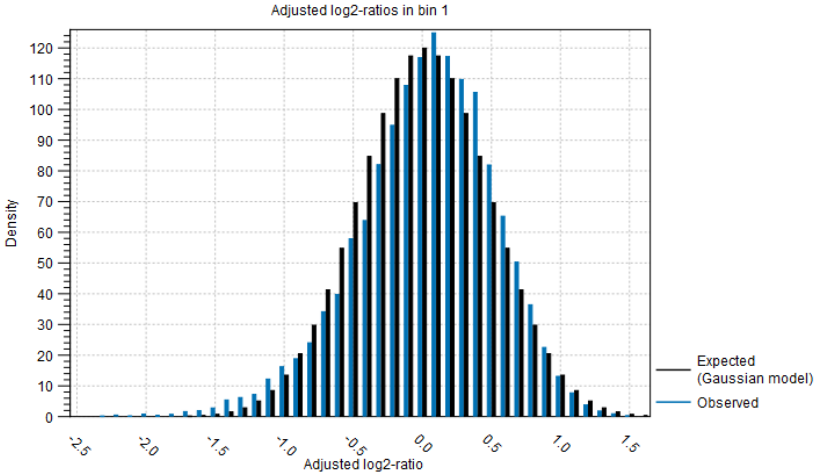

Figure 2.12: *An example bar chart from the algorithm report of the Copy Number Variation Detection plugin, showing how well the normal distribution assumption was fulfilled by the adjusted coverage log-ratios. Here, there is a good correspondence between the expected distribution and the observations.*

Prediction of region-level CNVs

The final section of the algorithm report is related to the region-level CNV prediction. In this part of the algorithm, the chromosomes are segmented into regions of similar adjusted mean log-ratios. More segments lead to a reduced variance per segment; in the extreme, where every target forms its own segment, the variance is zero. However, more segments also mean that the model contains more free parameters, and is therefore potentially over-fitted. A value known as the Bayesian Information Criterion (BIC) gives an indication of the balance of these two effects,

for any potential segmentation of a chromosome. The segmentation process aims to minimize the BIC, producing the best balance of accuracy and overfitting in the final segments.

The segmentation begins by identifying a set of potential breakpoints, known as local maximizers. The number of potential breakpoints at the start of the segmentation is shown in the "# local maximizers at start" column, and the corresponding BIC score is indicated in the "Start BIC" column. Breakpoints are removed strategically one-by-one, and the BIC score is calculated after each removal. When enough breakpoints have been removed for the BIC score to reach its minimum, the final number of breakpoints is shown in the "# local maximizers at end" column, and the corresponding BIC score is indicated in the "End BIC" column. A large reduction in the number of local maximizers indicates that it was possible to join many smaller CNV regions into larger ones.

Note: The segmentation process only produces regions of similar adjusted coverage log-ratios. Each segment is tested afterwards, to identify if it represents a CNV. Therefore, the number of segments shown in this table does not correspond to the number of CNVs actually predicted by the algorithm.

Chapter 3

Installation

The *Copy Number Variant Detection Plugin* is installed as a plugin. Plugins are installed using the plugin manager[1]:

Help in the Menu Bar | **Plugins and Resources... (** **)**

or **Plugins (** **) in the Toolbar**

The plugin manager has three tabs at the top:

- **Manage Plugins.** This is an overview of plugins that are installed.
- **Download Plugins.** This is an overview of available plugins on CLC bio's server.
- **Manage Resources.** This is an overview of resources that are installed.

To install a plugin, click the **Download Plugins** tab. This will display an overview of the plugins that are available for download and installation (see figure 3.1).

Clicking a plugin will display additional information at the right side of the dialog. This will also display a button: **Download and Install**.

Click the *Copy Number Variant Detection Plugin* and press **Download and Install**. A dialog displaying progress is now shown, and the plugin is downloaded and installed.

If the *Copy Number Variant Detection Plugin* is not shown on the server, and you have it on your computer (e.g. if you have downloaded it from our web-site), you can install it by clicking the **Install from File** button at the bottom of the dialog. This will open a dialog where you can browse for the plugin. The plugin file should be a file of the type ".cpa".

When you close the dialog, you will be asked whether you wish to restart the CLC Workbench. The plugin will not be ready for use until you have restarted.

[1] In order to install plugins on Windows, the Workbench must be run in administrator mode: Right-click the program shortcut and choose "Run as Administrator". Then follow the procedure described below.

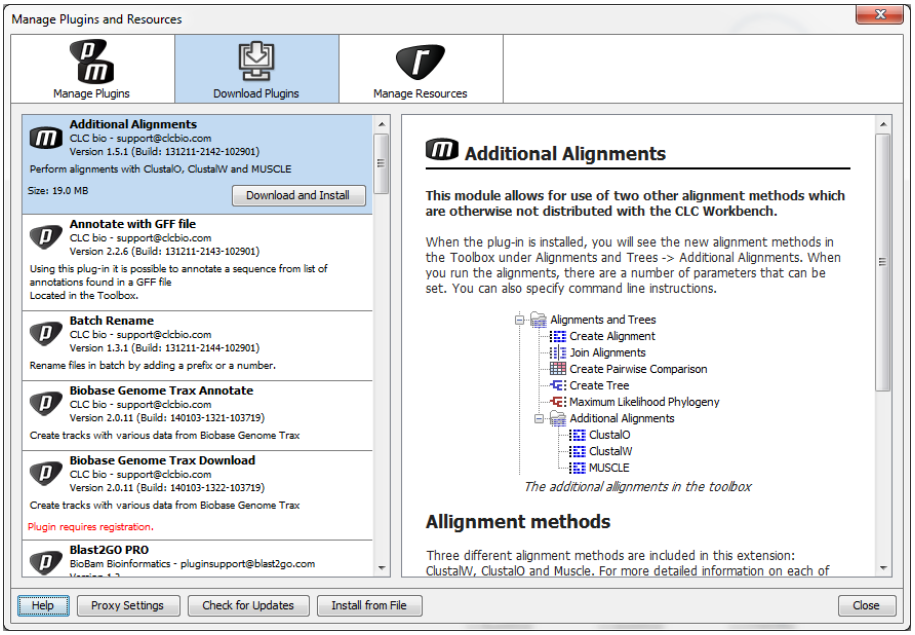
Manage Plugins and Resources
Manage Plugins
Download Plugins
Manage Resources
Additional Alignments
CLC bio - support@clcbio.com
Version 1.5.1 (Build: 131211-2142-102901)
Perform alignments with ClustalO, ClustalW and MUSCLE
Size: 19.0 MB
Download and Install
Annotate with GFF file
CLC bio - support@clcbio.com
Version 2.2.6 (Build: 131211-2143-102901)
Using this plug-in it is possible to annotate a sequence from list of annotations found in a GFF file
Located in the Toolbox.
Batch Rename
CLC bio - support@clcbio.com
Version 1.3.1 (Build: 131211-2144-102901)
Rename files in batch by adding a prefix or a number.
Biobase Genome Trax Annotate
CLC bio - support@clcbio.com
Version 2.0.11 (Build: 140103-1321-103719)
Create tracks with various data from Biobase Genome Trax
Biobase Genome Trax Download
CLC bio - support@clcbio.com
Version 2.0.11 (Build: 140103-1322-103719)
Create tracks with various data from Biobase Genome Trax
Plugin requires registration.
Blast2GO PRO
BioBam Bioinformatics - pluginsupport@blast2go.com
Additional Alignments
This module allows for use of two other alignment methods which are otherwise not distributed with the CLC Workbench.
When the plug-in is installed, you will see the new alignment methods in the Toolbox under Alignments and Trees -> Additional Alignments. When you run the alignments, there are a number of parameters that can be set. You can also specify command line instructions.
Alignments and Trees
Create Alignment
Join Alignments
Create Pairwise Comparison
Create Tree
Maximum Likelihood Phylogeny
Additional Alignments
ClustalO
ClustalW
MUSCLE
The additional allignments in the toolbox
Allignment methods
Three different alignment methods are included in this extension:
ClustalW, ClustalO and Muscle. For more detailed information on each of
Help
Proxy Settings
Check for Updates
Install from File
Close

Figure 3.1: *The plugins that are available for download.*

Chapter 4

Uninstall

Plugins are uninstalled using the plugin manager:

Help in the Menu Bar | **Plugins and Resources... (** **)**

or **Plugins (** **) in the Toolbar**

This will open the dialog shown in figure 4.1.

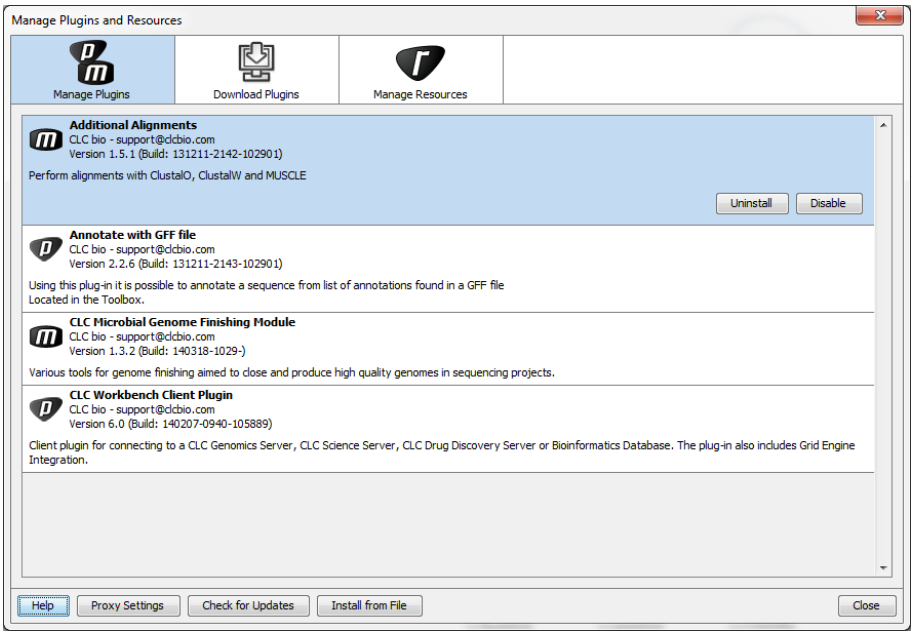
Manage Plugins and Resources
Manage Plugins
Download Plugins
Manage Resources
Additional Alignments
CLC bio - support@clcbio.com
Version 1.5.1 (Build: 131211-2142-102901)
Perform alignments with ClustalO, ClustalW and MUSCLE
Uninstall
Disable
Annotate with GFF file
CLC bio - support@clcbio.com
Version 2.2.6 (Build: 131211-2143-102901)
Using this plug-in it is possible to annotate a sequence from list of annotations found in a GFF file
Located in the Toolbox.
CLC Microbial Genome Finishing Module
CLC bio - support@clcbio.com
Version 1.3.2 (Build: 140318-1029-)
Various tools for genome finishing aimed to close and produce high quality genomes in sequencing projects.
CLC Workbench Client Plugin
CLC bio - support@clcbio.com
Version 6.0 (Build: 140207-0940-105889)
Client plugin for connecting to a CLC Genomics Server, CLC Science Server, CLC Drug Discovery Server or Bioinformatics Database. The plug-in also includes Grid Engine Integration.
Help
Proxy Settings
Check for Updates
Install from File
Close

Figure 4.1: *The plugin manager with plugins installed.*

The installed plugins are shown in this dialog. To uninstall:

Click the *Copy Number Variant Detection Plugin* | **Uninstall**

If you do not wish to completely uninstall the plugin but you don't want it to be used next time you start the Workbench, click the **Disable** button.

When you close the dialog, you will be asked whether you wish to restart the workbench. The plugin will not be uninstalled until the workbench is restarted.

Bibliography

[Li et al., 2012] Li, J., Lupat, R., Amarasinghe, K. C., Thompson, E. R., Doyle, M. A., Ryland, G. L., Tothill, R. W., Halgamuge, S. K., Campbell, I. G., and Gorringe, K. L. (2012). Contra: copy number analysis for targeted resequencing. *Bioinformatics*, 28(10):1307–1313.

[Niu and Zhang, 2012] Niu, Y. S. and Zhang, H. (2012). The screening and ranking algorithm to detect dna copy number variations. *Ann Appl Stat*, 6(3):1306–1326.